A Capacity Broker Architecture and Framework for Multi-tenant support in LTE-A Networks

Georgia Tseliou*, Konstantinos Samdanis†, Ferran Adelantado*, Xavier Costa Pérez† and Christos Verikoukis‡

* Open University of Catalonia (UOC), Barcelona, Spain

† NEC Europe Ltd., Heidelberg, Germany

‡ Telecommunications Technological Centre of Catalonia (CTTC), Castelldefels, Spain

{gtseliou, ferranadelantado}@uoc.edu, {Konstantinos.Samdanis, Xavier.Costa}@neclab.eu, cveri@cttc.es

***Abstract*—Resource provisioning in multi-operator scenarios requires an estimate of the tenants' traffic needs. This is necessary in the scenario where a Mobile Network Operator (MNO) owns the Radio Access Network (RAN) and many Mobile Virtual Network Operators (MVNOs) act as resellers of their host network's capacity under their own brands, to their own customers. In such scenarios, the forecasted MVNO traffic is the basis for providing resources suitable with the corresponding MVNOs demand. To that end, the dynamic provision of resources among MVNOs should be performed in flexible, short-term time scales. In this paper, we effectively address this issue by integrating the capacity broker into the 3rd Generation Partnership Project (3GPP) network management network architecture using the minimum set of enhancements. In addition, to fully exploit its capabilities, we propose the Multi-tenant Slicing (MuSli) of capacity algorithm, to allocate resources towards MVNOs in coarse time scales. MuSli considers the estimated capacity and the impact of the traffic type (i.e., guaranteed QoS and Best-Effort) in each MVNO, to provide better utilization of the host network's capacity. Our results highlight the gains in the number of served requests without compromising their service quality.**

I. Introduction

Mobile communications are entering a new era with the popularity of portable electronic devices, which gave rise to a plethora of new services with ever-increasing resource demands. Lately, Mobile Network Operators' (MNOs) revenues cannot keep pace, considering the cost to operate and upgrade their infrastructure. To date, operational observations show that there are underutilized resources, e.g., 50% of sites carry traffic that yields less than 10% of revenue [1]. Network sharing has been proposed to allocate these underutilized resources among Mobile Virtual Network Operators (MVNOs), providing another revenue source for MNOs. Studies have shown that it can recover up to 20% of operational costs for typical European MNOs and significantly reduce capital expenditures in developing countries (e.g., up to 70% in India) [2].

There are still many challenges to overcome, to achieve a viable network sharing business model appealing to MNOs. First, network sharing should be performed on demand, with resources acquired in the scale of minutes, while allocations are configured via signaling. A centralized resource management entity should facilitate this process. Its role is to assist the MNO owning a shared RAN (i.e., infrastructure provider), to fully exploit the unused capacity. The notion of this entity, referred to as capacity broker, has been introduced in the 3rd Generation Partnership Project (3GPP), from a business perspective [3]. Such a central entity is required to assure synchronization in resource sharing for such short-time scales, while satisfying Service Level Agreements (SLAs). Nevertheless, its integration into the 3GPP management architecture [4] is an open issue. In addition, a key question is how to exploit the functionality of capacity broker to accomplish an efficient resource allocation, by considering: (i) the global view of network resource utilization, and (ii) the knowledge of the expected traffic volumes, a challenging task due to lack of periodicity in short-term scale. Although many interesting studies on capacity slicing have been carried out, either they study the problem from different layer (e.g., [6]-[8]), or they introduce non-backwards compatible centralized entities with the existing 3GPP architecture (e.g., [9]).

To that end, the contributions of this paper concentrate on facilitating resource provisioning between MVNOs, by integrating the capacity broker in the 3GPP network management architecture with a minimum set of enhancements. Furthermore, to fully exploit its range of capabilities, we propose the Multi-tenant Slicing (MuSli) of capacity framework for on-demand resource allocation considering two types of traffic: (i) Guaranteed Quality of Service (QoS) with resources locked for explicit use by a MVNO and (ii) Best-Effort (BE) where resources are pooled and shared by all participants. To accomplish this, we follow a two-step approach: (i) we improve short-term forecasting techniques by extracting traffic variation trends and facilitate the capacity broker with accurate information regarding the expected traffic and (ii) we propose how to slice the available resources into these two types of traffic classes, depending on the forecasting and its respective accuracy.

The remainder of the paper is structured as follows. The related work is presented in Section II. In Section III we explain how the capacity broker is integrated in the 3GPP management architecture. Section IV introduces the system model along with the MuSli framework. Section V analyzes the simulation set-up and the evaluation results. Finally, Section VI concludes the paper.

II. State of the Art

The initial adoption of network sharing in 3GPP, concentrated on passive solutions, wherein MNOs share base station

sites, antennas, etc. Active sharing that followed, enabled operators to share network resources for long term periods according to contractual agreements. For active network sharing, 3GPP has specified two architectures in [5]: (i) the Multi-Operator Core Network (MOCN) and (ii) the Gateway Core Network (GWCN). In the former, each operator is sharing eNBs connected to core network elements belonging to each MNO using a separate S1 interface. In the latter, operators share additionally the Mobility Management Entity (MME). Our proposal is compatible with both 3GPP network sharing architectures, while introducing on-demand resource allocation via the means of signaling extensions of 3GPP network sharing management [4].

A preliminary approach for virtualizing an eNB is introduced in [6], by detailing the notion of hypervisor, that performs resource sharing among MNOs considering radio conditions, contracts and traffic load. In advancing the basic eNB virtualization, [7] introduces the Network Virtualization Substrate (NVS) that operates closely to the MAC scheduler. A tailored mixture of reserved and shared resources with respect to NVS component is proposed in [8], in order to flexibly allocate shared resources modifying the MAC scheduler. In this work, we adopt such NVS two-step process, but instead of concentrating on the MAC scheduler for performing resource differentiation, we leverage the capacity broker to provide different resource slices based on the expected traffic volume.

A study adopting the capacity broker paradigm in LTE is detailed in [9], regarding a range of capacity and spectrum sharing options. Unlike such an approach that introduces a new control plane interface to coordinate sharing agreements, our proposal is backwards compatible with the existing 3GPP network management architecture, reusing current interfaces, while introducing a minimum set of enhancements.

The accuracy of short-term load forecasts can significantly affect the capacity broker decisions for resource slicing. A wide range of solutions for short-term load forecasting have been reported in the literature [10], which can be distinguished in two categories. The first one employs characteristics of traffic loads, such as spatial/temporal relevance or self-similarity [11]. The second category employs techniques, such as exponential smoothing to study the intrinsic dimensionality [12], Kalman filtering to capture the evolution of traffic [13] or modern signal processing techniques such as compressive sensing [14]. In this paper, we investigate which of the above methods fits best the capacity broker paradigm and we provide a set of enhancements, to compensate the lack of periodicity and non-uniformities of a short-term prediction.

III. 3GPP Network Sharing Management Architecture

The overview of the 3GPP network sharing management architecture [4], in which we integrate the capacity broker and execute MuSli, is depicted in Fig. 1. The Master Operator-Network Manager (MO-NM) monitors the shared network via the Master Operator-Shared RAN-Domain Manager (MO-SR-DM) using Type 2 (i.e., Itf-N) interface. In turn, the latter

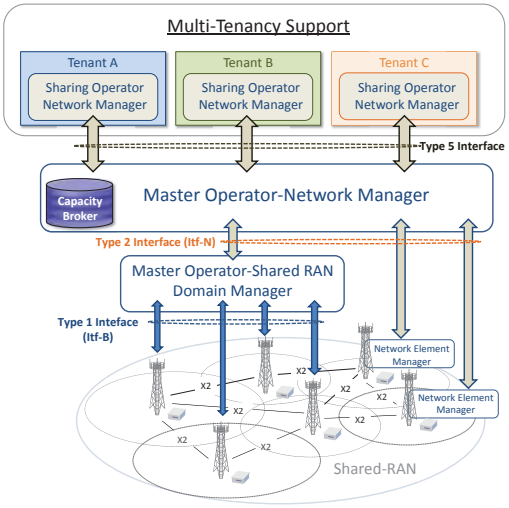

Fig. 1. Capacity Broker in 3GPP Network Sharing Management Architecture.

communicates with a set of shared base stations, via Type 1 (i.e., Itf-B) interface. All radio-related functions (i.e., Radio Resource Management, connectivity to core network etc.) take place in the level of the shared base stations. In addition, MO-NM enables the Sharing Operator-Network Manager (SO-NM), to monitor and control the allocated resources to MVNOs via Type 5 interface.

Given the existing architecture, we propose to place the capacity broker on the MO-NM, to facilitate the allocation of shareable resources, by automatic means and on an on-demand basis, to MVNOs. The capacity broker, by deciding which requests will be accepted, assures synchronization in resource sharing for short-time scales, while satisfying their SLAs. Thus, when co-locating it at MO-NM, it has rapid access to network monitoring information (such as Uplink/Downlink load and performance measurements), as well as to network planning information (i.e., MO-NM has collected this from MO-SR-DM). Then, the MO-NM uses the output of the capacity broker to inform the MO-SR-DM about which specific requests should be accepted and the shared base stations implement their respective radio-related functions. Our proposal requires extensions to Type 1, Type 2 and Type 5 interfaces. Type 1 and Type 2 need to accommodate the tenant identification (i.e., PLMN-id), resource allocation (e.g., Resource Blocks (RBs)), start time and duration of the request. In addition, Type 5, which is typically established upon an agreement, should include the list of MNO's cells involved in the capacity slicing process. All the above interfaces should support resource measurements and performance monitoring per MVNO. To that end, we introduce the PLMN-id within each corresponding packet. For the portion of pooled resources, monitoring information should be shared among all tenants' SO-NM systems.

IV. Multi-tenant Resource Slicing Framework

This section concentrates on elaborating a resource management framework, called Multi-tenant Slicing (MuSli), to

be executed in the capacity broker in coarse time-scales. Its objective is performing resource slicing among incoming requests considering two different traffic classes: guaranteed QoS and BE. The difference between the two aforementioned traffic classes lies in their distinct requirements in terms of radio resources. Thus, whereas guaranteed QoS traffic (usually identified with services such as voice) is characterized by a fixed transmission rate, BE traffic (identified, for instance, with data services) is defined in terms of average demanded data rate as well as more relaxed delay constraints.

In this scenario the management of the shared RAN resources, conducted by the capacity broker, has to deal with two main hurdles: i) the diversity of the traffic requests, and ii) the varying nature of the radio interface. Our methodology consists in using a forecasting procedure to predict the traffic volume in near future for all MVNOs considering the entire deployment and allocating resources with different quality to different traffic classes (e.g., for voice and data).

A. System Model

Let us define a scenario composed of a set of MVNOs, $\mathcal{V} = \{i : i = 0, \cdots, V\}$ sharing a single RAN. For the sake of simplicity, and without loss of generality, we assume hereafter that MVNO 0 is the owner of the shared RAN. The capacity broker (described in Section III) decides whether to accept or reject the incoming MVNOs' requests. Thus, it manages the shared RAN capacity to serve the capacity requests generated by the MVNOs in $\mathcal{V}$. In this context, the appropriate management of the available capacity is a twofold problem. First, the future capacity usage must be forecasted, and secondly the available expected capacity must be allocated to the set of received requests. According to the described traffic classes, the r^{th} request of the i^{th} MVNO can be defined as $g_{i,r}\{t_{i,r}, T_{i,r}, w_{i,r}\}$ for guaranteed QoS requests or as $b_{i,r}\{t_{i,r}, T_{i,r}, p_{i,r}, \lambda_{i,r}\}$ for BE requests, where $t_{i,r}$ is the request arrival time, $T_{i,r}$ is its duration, $w_{i,r}$ (in bps) is the requested transmission rate in guaranteed QoS traffic, $p_{i,r}$ is the average size of the packets (in bits/packet) and $\lambda_{i,r}$ is the average number of generated packets per second (both for BE traffic). It holds that each MVNO i generates a set of requests $\mathcal{R}_i = \{r : r = 1, \cdots, R\}$. With regard to the shared RAN, we consider a cellular deployment, consisting of a set of sectors $\mathcal{S} = \{s : s = 1, \cdots, S\}$. We denote by $x_{i,s}(t)$, the traffic volume of MVNO i in sector s at time t (expressed in RBs).

Upon the arrival of a request $r \in \mathcal{R}_i$ from MVNO $i \in \mathcal{V}$, the capacity broker must decide if the future availability of resources will suffice to serve the request r based on traffic forecasting. We define the column vector of the previous T_p+1 samples of $x_{i,s}(t)$ as $\mathbf{x^t_{i,s}} = (x_{i,s}(t - T_p), x_{i,s}(t - (T_p + 1)), \ldots, x_{i,s}(t))$, where t is expressed in minutes. Likewise, the vector of forecasted traffic volumes for the period $[t+1, t+T_f]$ is defined as $\hat{\mathbf{x}}^{\mathbf{t}}_{\mathbf{i,s}} = (\hat{x}_{i,s}(t+1), \hat{x}_{i,s}(t+2), \ldots, \hat{x}_{i,s}(t+T_f))$. Therefore, the forecasting function, f, can be defined as:

$$\begin{array}{rccc} f & : & \mathbb{R}^{T_p+1} & \longrightarrow & \mathbb{R}^{T_f} \\ & & \mathbf{x^t_{i,s}} & \longrightarrow & \hat{\mathbf{x}}^{\mathbf{t}}_{\mathbf{i,s}} \end{array} \quad (1)$$

There is a wide range of forecasting functions that could be used. In Section IV-C we propose some improvements to be applied to the forecasting function, and in Section V-B results obtained with different forecasting methods are evaluated.

Let us note, that the actual traffic volume can be seen as the forecasted traffic volume plus an error, i.e., $x_{i,s}(t) = \hat{x}_{i,s}(t) + \epsilon_{i,s}(t)$, with $\epsilon_{i,s}(t) \in \mathbb{R}$. Thus, in order to cope with the inaccuracy of the forecasted traffic, we define the Confidence Degree (CD) of the traffic volume of sector s, $\gamma^{\beta}_s(t)$, as the value that will not be exceeded by the actual traffic volume with probability β. Thus, it holds that

$$P[\hat{x}_s(t) + \epsilon_s(t) \leq \gamma^{\beta}_s(t)] = \beta, \quad (2)$$

where $\hat{x}_s(t) = \sum_{i \in \mathcal{V}} \hat{x}_{i,s}(t)$ and $\epsilon_s(t) = \sum_{i \in \mathcal{V}} \epsilon_{i,s}(t)$.

B. MuSli: Algorithm for Multi-tenant Slicing of Capacity

In our proposal, the capacity broker allocates to incoming guaranteed QoS requests, the RBs that are expected to be available based on the forecast traffic volume. Conversely, RBs with higher probability of being used, must be allocated to incoming BE requests. Note that the capacity broker defines the available capacity at time t in sector s and for a given β, as $C^{\beta}_s(t) = C - \gamma^{\beta}_s(t)$, where C is the total capacity of each sector (i.e., both $C^{\beta}_s(t)$ and C expressed as the number of RBs). Due to differences in the requirements of the two traffic classes, MuSli prioritizes guaranteed QoS requests over BE requests.

1) Guaranteed Requests: Let us consider a request $g_{i,r}\{t_{i,r}, T_{i,r}, w_{i,r}\}$ generated by MVNO i to serve a specific user. This user moves around the scenario with a trajectory described by $\mathcal{M}_{i,r} = \{(s_1, \tau_1), \cdots, (s_M, \tau_M)\}$, where the tuple (s_m, τ_m) refers to the m^{th} sector visited by the user ($s_m \in \mathcal{S}$) and the time at which the user enters sector m (i.e., $\tau_m \in [t_{i,r}, t_{i,r} + T_{i,r}]$). For this specific case, the capacity broker should only accept the request if the transmission rate (i.e., $w_{i,r}$ bps), can be guaranteed along $T_{i,r}$. In other words, it would be accepted if

$$\min_{t \in [\tau_m, \tau_{m+1})} \left\{ C^{\beta}_{s_m}(t) \right\} \geq \frac{w_{i,r}}{w_{s_m}}, \ \forall (s_m, \tau_m) \in \mathcal{M}_{i,r}, \quad (3)$$

where w_{s_m} is the average transmission rate per RB, within sector s_m. Yet, as trajectories are unknown by the capacity broker, the acceptance/rejection decision is performed stochastically. We assume, that at time t_0 a set of new guaranteed traffic requests, namely $\mathcal{G}(t_0)$, reaches the capacity broker. According to the data collected until t_0, the probability that the new traffic will be served by sector s can be calculated as:

$$\alpha_s = \frac{w_s \sum_{i \in \mathcal{V}} ||\mathbf{x^{t_0}_{i,s}}||_1}{\sum_{s' \in \mathcal{S}} w_{s'} \sum_{i \in \mathcal{V}} ||\mathbf{x^{t_0}_{i,s'}}||_1}, \quad (4)$$

where $||\cdot||_1$ stands for the 1-norm operand. Initially, the set of accepted requests is empty and denoted by $\mathcal{G}'(t_0) = \emptyset$. Thus, a request $g_{i,r}\{t_0, T_{i,r}, w_{i,r}\} \in \mathcal{G}(t_0)$ is accepted if $F_g(g_{i,r}) \geq 0$ for $\forall t \in [t_0, T_{i,r}]$, where $F_g(g_{i,r})$ yields the available RBs given that $g_{i,r}$ is accepted. Hence, it is expressed as:

$$F_g(g_{i,r}) = \sum_{s\in\mathcal{S}} \alpha_s \left[C_s^{\beta}(t) - \left(\sum_{g_{j,k}\in\mathcal{G}'(t)} \frac{w_{j,k}}{w_s} \right) - \frac{w_{i,r}}{w_s} \right]. \quad (5)$$

We calculate (5) for all sectors of the deployment (each one weighted by α_s), by subtracting the resources that are needed to serve the already accepted requests and the resources required for the incoming $g_{i,r}$, from the available capacity of sector s in time t. If accepted, $g_{i,r}$ is removed from $\mathcal{G}(t_0)$ and it is included in $\mathcal{G}'(t_0)$. This procedure is repeated for all requests in $\mathcal{G}(t_0)$.

2) Best Effort Requests: BE requests are served after accommodating the guaranteed ones. However, since these requests do not have the strict data rate constraint imposed by the latter, the capacity broker can allocate them resources more flexibly. Let us consider that at time t_0, a set of new BE traffic requests (i.e., $\mathcal{B}(t_0)$), reaches the capacity broker.

For a given request $b_{i,r}\{t_0, T_{i,r}, p_{i,r}, \lambda_{i,r}\} \in \mathcal{B}(t_0)$, the average amount of bits generated along its duration (i.e., $T_{i,r}$), may be expressed as $T_{i,r}p_{i,r}\lambda_{i,r}$ bits. Following the same rationale stated in Section IV-B1, the average number of RBs required to serve this request in sector s, is equal to $\frac{T_{i,r}p_{i,r}\lambda_{i,r}}{w_s T_{sf}}$, where T_{sf} is the sub-frame time of LTE-A (i.e., 0.5 msec). However, the service disruption tolerance of BE traffic allows the capacity broker to allocate resources more elastically. Therefore, if we define the set of accepted new BE requests at time t_0 as $\mathcal{B}'(t_0)$, which is initially empty (i.e., $\mathcal{B}'(t_0) = \emptyset$), a request $b_{i,r}\{t_0, T_{i,r}, p_{i,r}, \lambda_{i,r}\}$ will only be accepted if $F_b(b_{i,r}) \geq 0$. $F_b(b_{i,r})$ expresses the available RBs given that $b_{i,r}$ is accepted and it is expressed as

$$\begin{aligned} F_b(b_{i,r}) \;=\; & \sum_{s\in\mathcal{S}} \alpha_s \left[\int_{t_0}^{t_0+T_{i,r}} \left(C_s^{\beta}(t) - \sum_{g_{j,k}\in\mathcal{G}'(t)} \frac{w_{j,k}}{w_s} \right) dt - \right. \\ & \left. - \left(\sum_{b_{j,k}\in\mathcal{B}'(t)} \frac{\lambda_{j,k}p_{j,k}T_{j,k}}{w_s T_{sf}} \right) - \frac{\lambda_{i,r}p_{i,r}T_{i,r}}{w_s T_{sf}} \right]. \quad (6) \end{aligned}$$

We compute (6), by subtracting the required resources to serve the already accepted BE requests and the resources to serve $b_{i,r}$, from the available capacity in sector s, along the duration of the request (i.e., $T_{i,r}$). As guaranteed requests precede, the available sector capacity for BE requests is calculated by deducing the resources needed to serve the accepted guaranteed ones. If request $b_{i,r}$ is accepted, then it is removed from $\mathcal{B}(t_0)$ and it is included in $\mathcal{B}'(t_0)$.

C. Capacity Forecasting

The flexibility of the network sharing management architecture (i.e., detailed in Section III), required to provide short-time scale dynamic provision of resources, poses challenges into traffic forecasting. There are several factors that affect the variation of the traffic along time, such as the mobility of the users, the deployment of the eNBs, etc. In our work, non-uniformities in the prior traffic load are due to gravity points of the mobility model. Given that the time horizon of the forecasting (which is taken into account by the capacity broker to make admission decisions) depends on $T_{i,r}$ of each request, we propose the prior decoupling of the variation trends that exist in $\mathbf{x}^{\mathbf{t}}_{\mathbf{i,s}}$.

In order to conduct the decoupling, the forecasting function, first defined in (1), performs the Fast Fourier Transform (FFT) of the traffic vector for each sector, i.e. $\mathbf{X_{i,s}} = \mathcal{F}\{\mathbf{x}^{\mathbf{t}}_{\mathbf{i,s}}\} = \{X_{i,s}(k) : k = 0, \ldots, T_p\}$, where $\mathcal{F}\{\cdot\}$ stands for the FFT transform. After applying the FFT, the capacity broker identifies the set of peaks of $\mathbf{X_{i,s}}$ and then splits it up into a set of components. Hence, for the jth peak of $\mathbf{X_{i,s}}$, located at $k = k_j$, we define $\mathbf{X}^{\mathbf{j}}_{\mathbf{i,s}} = \{X^j_{i,s}(k) : k = 0, \ldots, T_p\}$ where $X^j_{i,s}(k) = \{X_{i,s}(k) \cdot \Lambda_j(k) : k = 0, \ldots, T_p\}$, with $\Lambda_j(k) = 1$ for $k_{j,min} < k < k_{j,max}$ and $\Lambda_j(k) = 0$ otherwise. If a minimum threshold X_{min} is set, the limits $k_{j,min}$ and $k_{j,max}$ are defined as $k_{j,min} = (k_{j-1} + k_j)/2$ and $k_{j,max} = (k_j + k_{j+1})/2$. Finally, the decoupled traffic is generated as $\mathbf{x}^{\mathbf{t,j}}_{\mathbf{i,s}} = \mathcal{F}^{-1}\{\mathbf{X}^{\mathbf{j}}_{\mathbf{i,s}}\}$, where $\mathcal{F}^{-1}\{\cdot\}$ is the Inverse Fast Fourier Transform (IFFT).

The important point to note here, is that each $\mathbf{x}^{\mathbf{t,j}}_{\mathbf{i,s}}$ isolates a component of the traffic variation, and therefore it can be the basis for a more accurate forecasting. Thus, for a given forecasting method $f_{FM} : \mathcal{R}^{T_p+1} \rightarrow \mathcal{R}^{T_f}$, the forecasted vector of sector s assuming that J peaks are identified in $\mathbf{X_{i,s}}$ may be expressed as: $\mathbf{x}^{\mathbf{t}}_{\mathbf{i,s}} = \sum_{j=1}^{J} f_{FM}\left(\mathbf{x}^{\mathbf{t,j}}_{\mathbf{i,s}}\right)$.

In Section V-B, results for different f_{FM} are obtained, i.e., ARIMA, compressive sensing-based method, Kalman Filter and Holt-Winters.

D. Forecasting Error and Confidence Degree

As stated in (2), the forecasting error and the CD are tightly coupled. Specifically, the error $\epsilon_{i,s}(t)$ depends on t, T_p, T_f and f_{FM}. Therefore, in Section V the error (and consequently the CD, γ_s^{β}) is estimated empirically by applying the following methodology:

- 1000 realizations of $\epsilon_{i,s}(t)$ are collected (i.e., in a deployment with differently loaded cells) for each forecasting method. Next the 1000 sample measurements are used to obtain the empirical density function by employing the Kernel Density Estimation Technique (KDE) [15]. KDE is a non-parametric method, and thus it is not necessary to make assumptions on the $\epsilon_{i,s}(t)$ distribution.
- For computing the CD, a profile of 1000 experimentally estimated capacity values (i.e., $\hat{x}_{i,s}(t)$) is created. This profile is used as an observation. As previously, the KDE is used to obtain the empirical density function.

V. Performance Evaluation

A. Scenario and Parameters

We consider an Urban Micro-cell scenario consisting of 19 BSs with 3 sector antennas each one (total $S = 57$ sectors), based on the IMT-Advanced evaluation guidelines [16]. Table I summarizes the detailed system parameters. Users move in the network following the SLAW model, which is a human walk mobility model, considering mobiles moving in confined

gravity areas [17]. With regard to the forecasting, we collected the prior data traffic records from 57 sectors with coverage 2000 m^2. Each data record contains: *Time*, *Sector ID* and *RBs*. For our simulations, we use two traffic models to represent guaranteed QoS and BE traffic following parameters in [18]. The users generate guaranteed Constant Bit Rate (CBR) VoIP traffic with transmission rate 64 Kb/s, as well as BE traffic FTP requests with file size 0.5 Mbyte every 60 seconds. The inter-arrival rate follows a Poisson distribution.

TABLE I
BASIC SYSTEM PARAMETERS USED IN THE SIMULATION

Parameters	**Settings/Assumptions**
Network layout	19 BSs ($S = 57$ sectors)
Tenants	$V = 2$ (MNO:$i = 0$ and MVNOs:$i = 1, 2$)
Inter-site distance	200 m (ISD)
Bandwidth	20 MHz (100 RBs) 2.5 GHz
Path loss Model	$36.7log10$(d[m]) + $22.7+26log10$(fc[GHz])
Shadow fading	Lognormal, μ = 0, std.=4 dB

B. Forecasting Evaluation

For our study, we examine the following short-term capacity forecasting methods: ARIMA [11], compressive sensing-based method [14], Kalman filter [13], and Holt-Winters [12]. To identify the most suitable method for the capacity broker, we generated data that spanned in a two-hour prior time period ($T_p = 120$ minutes) using SLAW mobility model [17] and we obtained a $T_f = 20$ minute forecast. According to SLAW, the generated data capture spatial non-uniformities due to variations in users' trajectories. To compare the performance of the above methods, we consider a set of network instances with different load conditions. We use Root Mean Square Error (RMSE) to measure the forecasting accuracy of the studied methods. RMSE represents the sample standard deviation of the difference between predicted and observed values. The results in Table II show that the most accurate forecast (in the sense of minimizing RMSE) is the Holt-Winters technique. Applying the decoupling method of Section IV-C (i.e., FFT), outperforms the case of forecasting the prior traffic vector without any decomposition. The highest gain is achieved in methods that leverage the seasonality of the input data (i.e., Holt-Winters and Kalman Filter).

TABLE II
RMSE OF THE STUDIED FORECASTING METHODS

	HW	Kalman	Comp.Bas.Sens.	Arima
Without FFT	4.18	5.25	7.1	9.9
With FFT	2.46	3.97	5.96	7.43

C. MuSli Results

In this section we study the performance of the capacity broker, by executing MuSli for varying forecasting CDs (i.e., where $\beta = \{90\%, 95\%, 99\%\}$). The capacity slicing is applied by considering all network cells. In our scenario, MVNOs generate both guaranteed QoS and BE requests, with a traffic mix ratio 20% - 80%. We study different parameters for the time duration of the prediction (i.e., T_f), while augmenting

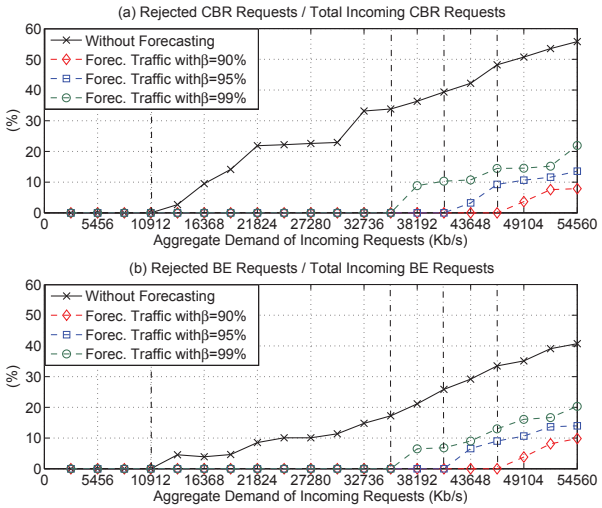

Fig. 2. (a) Rejected Guaranteed Requests and (b) Rejected BE Requests.

the aggregate demand of incoming requests. At the arrival moment of a request (i.e., t_0), MuSli decides which requests to accept/reject by checking the CD of the prediction. To evaluate its performance, we compare it with the baseline scenario, where admission for an incoming request is based on resource availability at t_0. We conducted Monte-Carlo event-based simulations in MATLAB® with 1000 iterations to achieve statistical validity for each forecasting step.

1) Admission of Incoming Requests: We begin the evaluation of MuSli by emphasizing the effect of slicing the overall capacity using various CDs, on the number of accepted/rejected requests. Fig. 2 depicts the percentages of (a) rejected guaranteed QoS (i.e., CBR) and (b) BE (i.e., FTP) requests. In general, when the capacity broker applies MuSli with different CDs, more requests are accepted compared with the baseline scheme. Even for the case of MuSli with $\beta = 99\%$ for 46376 Kb/s aggregate demand (i.e., the most conservative approach in slicing resources), the capacity broker rejects 10.28% of the incoming guaranteed requests whereas the baseline scenario 39.34%. In particular, we observe that the capacity broker that applies MuSli with high β rejects more requests, since it considers less capacity to allocate. The vertical dashed lines denote the limit of offered load that can be accepted without any rejection (i.e., 10912 Kb/s for the baseline scheme, 35464 Kb/s for MuSli with $\beta = 99\%$, 40920 Kb/s for MuSli with $\beta = 95\%$ and 46376 Kb/s for MuSli with $\beta = 90\%$).

In principle, there is a trade-off between service quality assurance and number of served requests. On the safe side, using high β on the predicted traffic, ensures service quality but results into accepting fewer requests. Therefore, the capacity broker can tune the CD of the forecasting, to treat requests, according to the desired level of certainty in assuring service quality. For this reason, in Fig. 2, the capacity broker that applies MuSli with high β rejects more both guaranteed and BE requests compared with MuSli with lower β. Moreover, when comparing Fig. 2(a) and Fig. 2(b), BE requests are

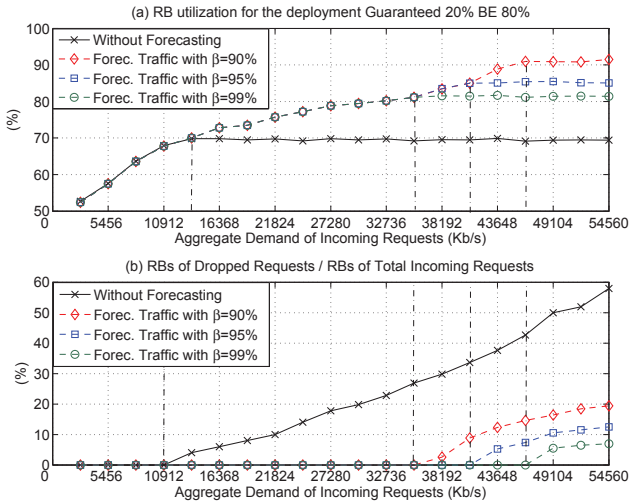

Fig. 3. (a) RB utilization and (b) SLA violation.

rejected with lower probability compared to guaranteed ones. This is due to their more relaxed delay constraints.

2) Resource Block Utilization: In Fig. 3, we study the percentage of (a) RB utilization and (b) RBs of dropped requests, versus their aggregate demand. In our scenario, a guaranteed request is dropped when it lacks resources at some point along its duration, whereas a BE request is dropped when its total transmission time is higher than a threshold time [18]. Given that both these cases result into disregarding the agreed SLA, let us refer to them as SLA violation.

In Fig. 3(a), we observe that for low incoming demand (up to 10912 Kb/s), accepting requests based only in current resource knowledge (i.e., baseline approach) results into the same utilization as the one achieved by the capacity broker. As soon as the baseline approach starts rejecting the incoming demand (i.e., starting at 13640 Kb/s as shown in Fig. 2), the RB utilization stabilizes around 69.8%. However, traffic prediction can prove to be very useful for higher demands. The capacity broker, by applying MuSli improves the utilization of the network. All RB utilization curves stabilize at a certain offered load limit, beyond which the capacity broker rejects requests (as also depicted in Fig. 2). As we expected, applying MuSli with high β results in restricted utilization compared to MuSli with lower β. As shown in Fig. 2, when using high β more requests are rejected and thus the RB utilization is limited. Since we are considering the whole deployment, particular overloaded cells (i.e., gravity points of the mobility model) restrict the available resources that the capacity broker can allocate in the slicing process.

Fig. 3(b) illustrates the percentage of RBs of dropped requests due to violation of the SLA. Although MuSli with high β rejects more requests (see Fig. 2), it is less likely to have dropped ones (e.g., when the real traffic is higher than the chosen CD). For instance, for 43648 Kb/s, an operator can choose Musli with $\beta = 90\%$ to achieve 90% utilization in the cost of having 11% SLA violation. On the contrary, being more conservative and choosing MuSli with $\beta = 99\%$, will result into 81% utilization without any SLA violation. This confirms the trade-off between service quality assurance and number of served requests.

VI. Conclusion

In this paper, we integrated the capacity broker in the 3GPP management architecture with a minimum set of enhancements. In addition, by leveraging traffic non-uniformities in a shared deployment, we proposed MuSli, a framework to be implemented by the capacity broker in coarse time scales. Along with our proposal, we introduced a decoupling process to extract variation trends in irregular traffic patterns and improve traffic forecasting. MuSli, by deciding how to slice the deployment's capacity among two types of requests (i.e., Guaranteed QoS and BE), improves network's performance by (i) increasing the accepted requests, and (ii) decreasing the underutilized resources. Our results can be leveraged by infrastructure owners, to flexibly allocate capacity to tenants, considering different types of services and the uncertainty of expected traffic. In our future work, we are planning to further study the degree of certainty in resource provisioning, based on the density of the deployment and the variation of mobility.

Acknowledgement

This work has been funded by the MITN Project CROSSFIRE (PITN-GA-2012-317126).

References

[1] K. Larsen, "Network Sharing Fundamentals," Jul. 2012.
[2] GSMA, "Network Infrastructure Sharing," Sep. 2012.
[3] *3GPP TR 22.852, Study on RAN Sharing enhancements, Rel.12*, Sept. 2014.
[4] *3GPP TS 32.130 Telecommunication management; Network Sharing; Concepts and requirements, Rel.12*, Dec. 2014.
[5] *3GPP TS 23.251, Network Sharing; Architecture and Functional Description*, Rel.13, Mar. 2015.
[6] Y. Zaki *et al.*, "LTE Wireless Virtualization and Spectrum Management," in *IFIP WMNC, Budapest*, Oct. 2010.
[7] X. Costa-Perez *et al.*, "Radio Access Network Virtualization for Future Mobile Carrier Networks," *IEEE Comm. Mag.*, vol. 51, no. 7, Jul. 2013.
[8] T. Guo and R. Arnott, "Active LTE RAN Sharing with Partial Resource Reservation," in *IEEE VTC Fall, Las Vegas*, Sep. 2013.
[9] J. Panchal, R. Yates, and M. Buddhikot, "Mobile network resource sharing options: Performance comparisons," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, Sep. 2013.
[10] W. Jiewu *et al.*, "User traffic collection and prediction in cellular networks: Architecture, platform and case study," in *IEEE IC-NIDC, Beijing*, Sep. 2014.
[11] Y. Shu *et al.*, "Wireless traffic modeling and prediction using seasonal arima models," in *IEEE ICC, Anchorage*, vol. 3, May 2003.
[12] D. Tikunov and T. Nishimura, "Traffic prediction for mobile network using Holt-Winter's exponential smoothing," in *15th SoftCOM, Dubrovnik*, Sep. 2007.
[13] A. Yadav *et al.*, "A constant gain Kalman filter approach to target tracking in wireless sensor networks," in *IEEE ICIIS, Chennai*, Aug. 2012.
[14] R. Li *et al.*, "Energy savings scheme in radio access networks via compressive sensing-based traffic load prediction," *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 4, Apr. 2014.
[15] J. Hwang *et al.*, "Nonparametric multivariate density estimation: a comparative study," in Signal Processing, IEEE Transactions on , vol.42, no.10, pp.2795-2810, Oct 1994
[16] ITU-R, "Guidelines for evaluation of radio interface technologies for IMT-Advanced," Report ITU-R M.2135-1, Dec. 2009.
[17] K. Lee *et al.*, "SLAW: Self-similar Least-action Human Walk," *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, Apr. 2012.
[18] *3GPP TR 36.814 Further advancements for E-UTRA physical layer aspects*, Rel. 9, Mar. 2010.